Institut Supérieur d'Informatique

DAKAR

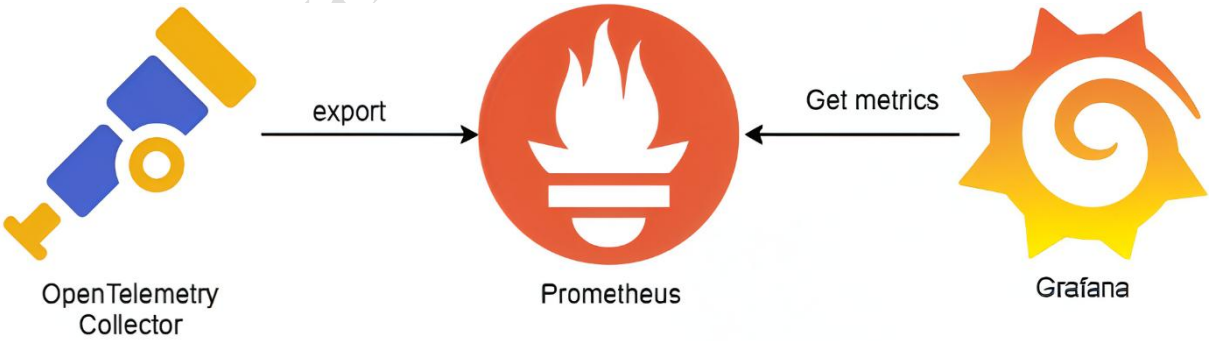
Département Réseaux et Systèmes

Spécialité : Sécurité des Systèmes d'Informations et Monétiques

Module : Administration et supervision des Services réseaux Avancés (IP Services)

Supervisé par : M. Massamba LO

Rapport de la mise en place de Prometheus, OpenTelemetry et Grafana



Réalisé par :

Khadim Rassoul THIAM

Khadimrassoulthiam10@gmail.com

Année Académique 2024 – 2025

ISI DAKAR

Sommaire

I. Introduction à Prometheus, OpenTelemetry et Grafana

II. Problématique

III. Avantages de la combinaison Prometheus / OpenTelemetry / Grafana

IV. Limites

V. Comparaison avec d'autres solutions de monitoring

VI. Installation et configuration de Prometheus, OpenTelemetry et Grafana sur Ubuntu (serveur)

VII. Installation et configuration des agents sur Windows

VIII. Installation et configuration des agents sur Debian

IX. Intégration avec Grafana

X. Analyse des données collectées

XI. Conclusion

Institut Supérieur d'Informatique - ISI DAKAR

I. Introduction à Prometheus, OpenTelemetry et Grafana

Prometheus est un système de supervision open-source spécialisé dans la collecte de métriques, développé initialement par SoundCloud. Il stocke les données sous forme de séries temporelles et fournit un langage de requête puissant (PromQL) pour l'analyse et la corrélation des métriques.

OpenTelemetry est une initiative standardisée qui fournit des API, SDK et outils pour collecter de la télémétrie (métriques, logs, traces) depuis n'importe quelle application ou infrastructure. C'est une solution agnostique, largement adoptée par les éditeurs et la communauté DevOps.

Grafana est une plateforme de visualisation et d'analyse de données capable de se connecter à de nombreuses sources (Prometheus, InfluxDB, Elasticsearch, etc.) et de produire des tableaux de bord dynamiques, personnalisés et interactifs.

La combinaison de ces trois outils offre un écosystème complet pour l'observation des systèmes.

II. Problématique

Les systèmes informatiques actuels sont complexes, distribués et soumis à des charges variables. Les incidents de performance ou de disponibilité peuvent avoir un impact direct sur la continuité des services. Les outils classiques de supervision ne permettent pas toujours de :

- Collecter en temps réel des métriques précises.
- Corréler efficacement métriques, logs et traces.
- Offrir une visualisation flexible et compréhensible par les équipes.

La combinaison Prometheus / OpenTelemetry / Grafana répond à cette problématique en offrant une solution intégrée, modulaire et extensible.

Institut Supérieur d'Informatique - ISI DAKAR

III. Avantages de la combinaison

1. **Open-source et gratuit** : pas de coûts de licence.
2. **Collecte complète** : prise en charge des métriques, logs et traces.
3. **Performance** : ingestion rapide de millions de points de données.
4. **Visualisation avancée** : Grafana offre des tableaux de bord personnalisés.
5. **Interopérabilité** : OpenTelemetry fonctionne avec une multitude de langages et plateformes.
6. **Alertes flexibles** : configuration d'alertes dans Prometheus ou Grafana.
7. **Scalabilité** : adaptation aux petites comme aux très grandes infrastructures.

IV. Limites

1. **Mise en place complexe** : nécessite la maîtrise de plusieurs outils.
2. **Ressources matérielles** : serveurs puissants nécessaires pour un grand volume de données.
3. **Courbe d'apprentissage importante** : surtout pour la configuration avancée.

V. Comparaison avec d'autres solutions

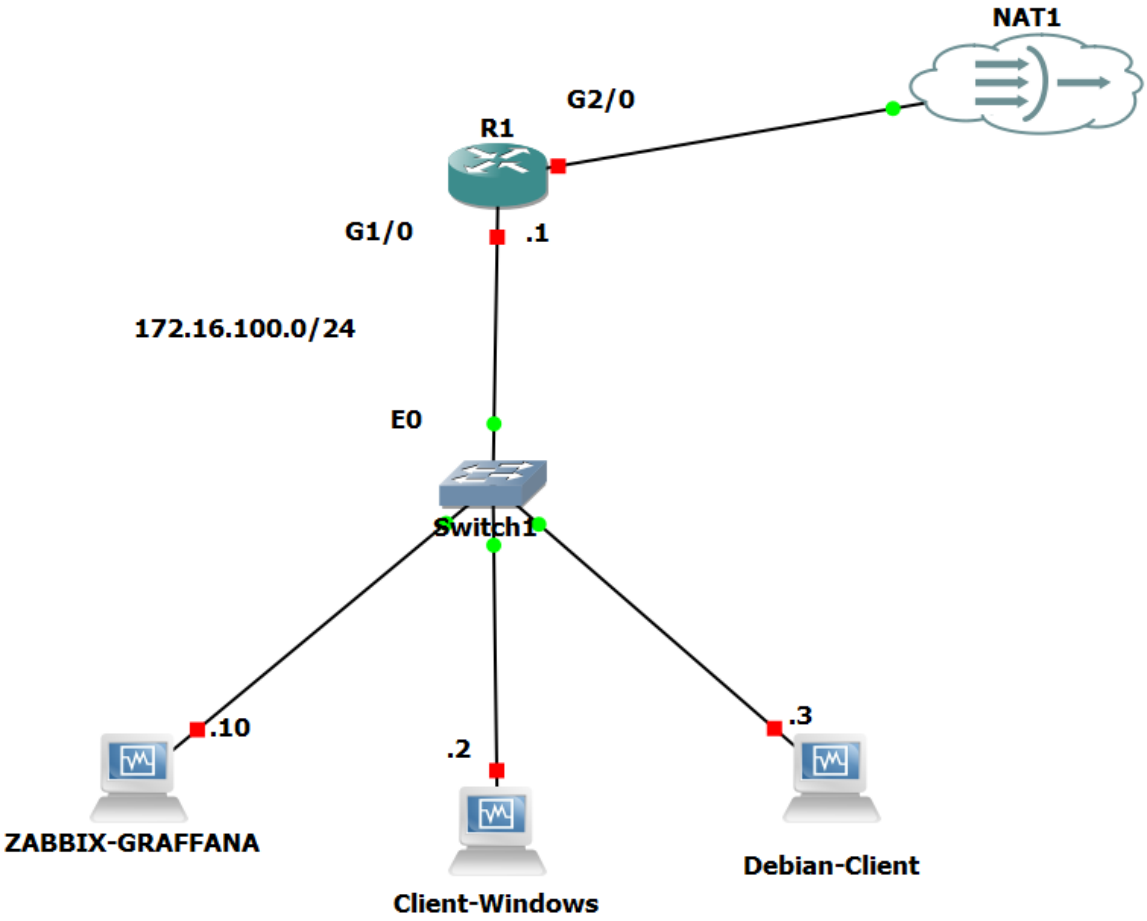
Outil / Stack	**Avantages**	**Inconvénients**
ELK Stack	Excellente analyse de logs	Moins performant pour métriques temps réel
Zabbix	Supervision complète, alertes	Moins adapté aux traces applicatives
Datadog	Solution tout-en-un	Payant
Prometheus + OpenTelemetry + Grafana	Flexible, puissant, open-source	Complexité de déploiement

---

Institut Supérieur d'Informatique - ISI DAKAR

VI. Installation et configuration de Prometheus et OpenTelemetry sur Ubuntu (serveur)

Topologie



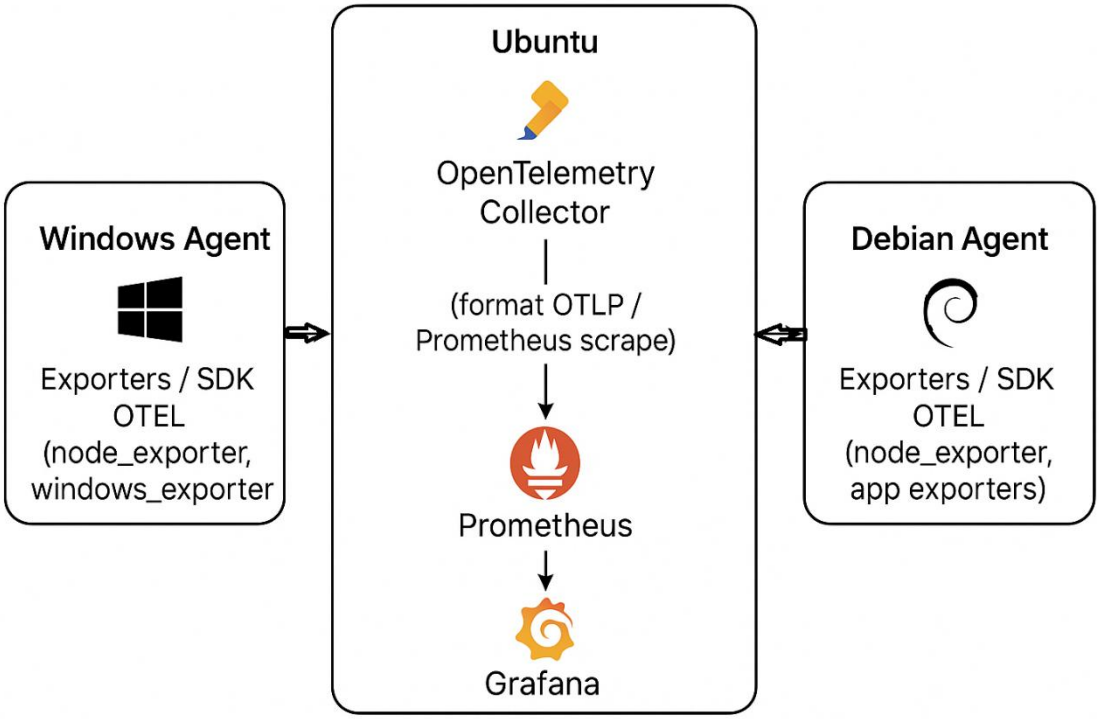
Institut

Logique

Rôle de chaque composant

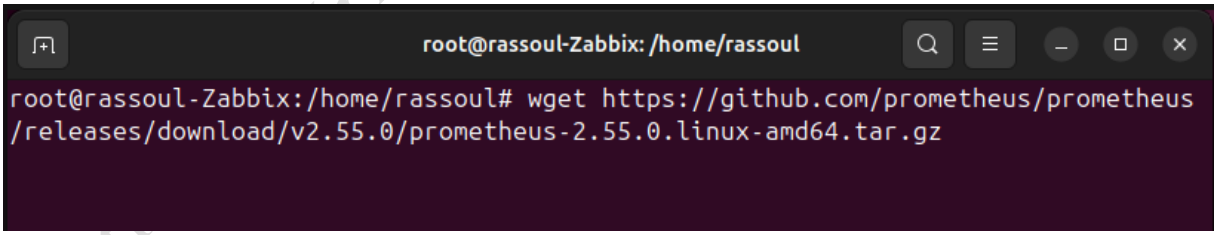
- **Windows Agent**
  - Installe **windows_exporter** pour envoyer métriques système (CPU, RAM, disque, réseau).
  - Utilise éventuellement un **OTEL SDK** ou **OTEL Agent** pour transmettre métriques/logs applicatifs.
  - Envoie soit directement vers le **OTEL Collector** sur Ubuntu, soit exposé en endpoint HTTP pour que Prometheus scrape.
- **Debian Agent**
  - Installe **node_exporter** (métriques système).
  - Envoie également via OTEL Collector ou endpoint HTTP.
  - Peut aussi envoyer des logs vers l'OTEL Collector via protocoles OTLP/HTTP ou gRPC.
- **Ubuntu (Serveur Central)**
  - **OpenTelemetry Collector** :
    - Réceptionne les données des deux agents.
    - Transforme (pipeline processors) et exporte vers Prometheus.
  - **Prometheus** :
    - Scrape les métriques exposées par OTEL Collector ou directement par les agents.
    - Stocke ces métriques en base interne TSDB (Time Series Database).
  - **Grafana** :
    - Se connecte à Prometheus comme datasource.
    - Crée tableaux de bord (dashboards) pour visualiser les données collectées.

Institut Supérieur d'Informatique - ISI DAKAR

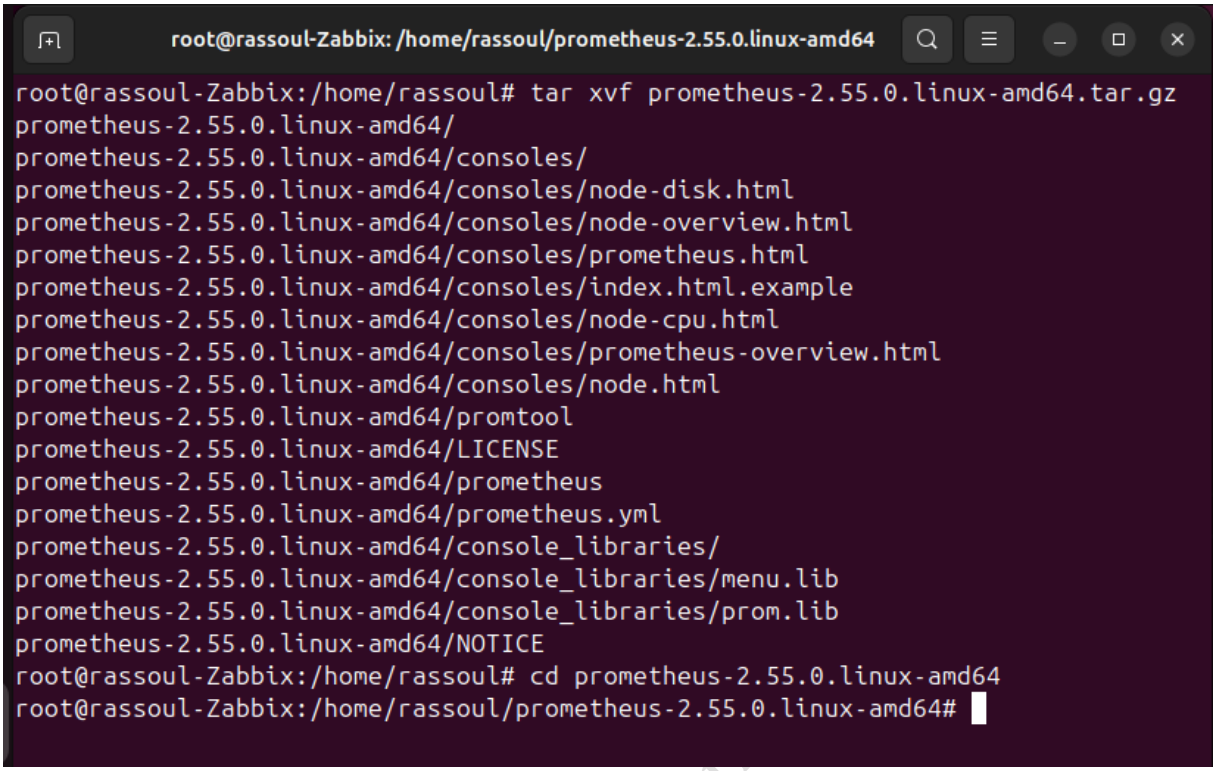


Installation de Prometheus

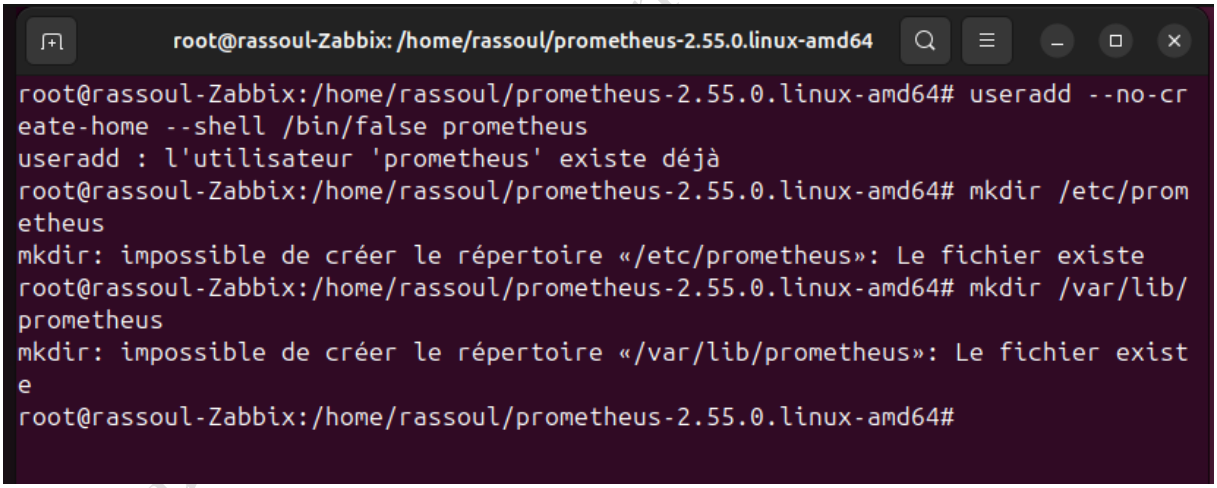
Télécharger le fichier d'installation



Désarchiver le fichier



Ajouter l'utilisateur **Prometheus** avec un **shell null**



Copier les dossiers suivants et attribuer les droits

```
root@rassoul-Zabbix: /home/rassoul/prometheus-2.55.0.linux-amd64
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# cp prometheus p
romtool /usr/local/bin/
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# cp -r consoles/
 console_libraries/ /etc/prometheus/
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# cp prometheus.y
ml /etc/prometheus/
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# chown -R promet
heus:prometheus /etc/prometheus /var/lib/prometheus
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# chown prometheu
s:prometheus /usr/local/bin/prometheus /usr/local/bin/promtool
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64#
```

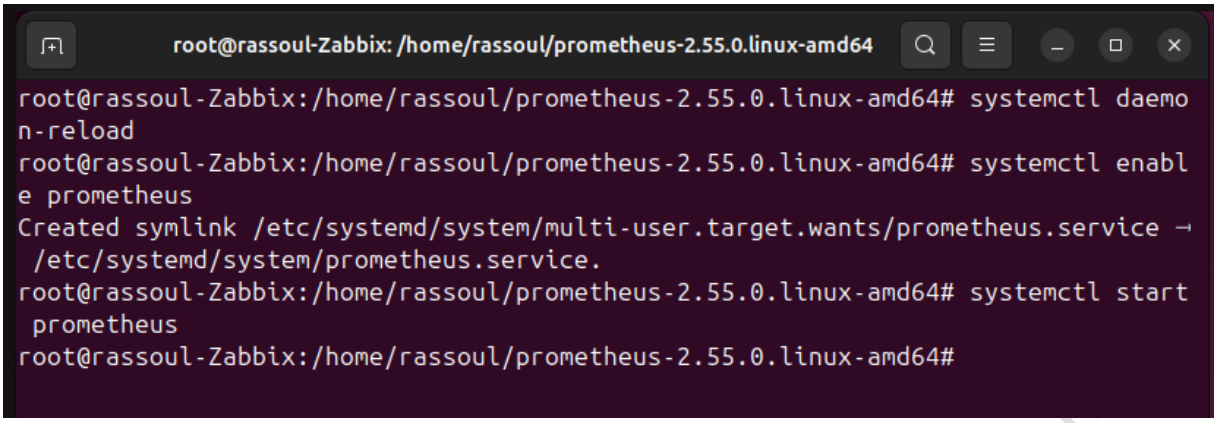
Créer le fichier de gestion du service prometheus

```
root@rassoul-Zabbix: /home/rassoul/prometheus-2.55.0.linux-amd64
root@rassoul-Zabbix:/home/rassoul/prometheus-2.55.0.linux-amd64# vim /etc/system
d/system/prometheus.service
```

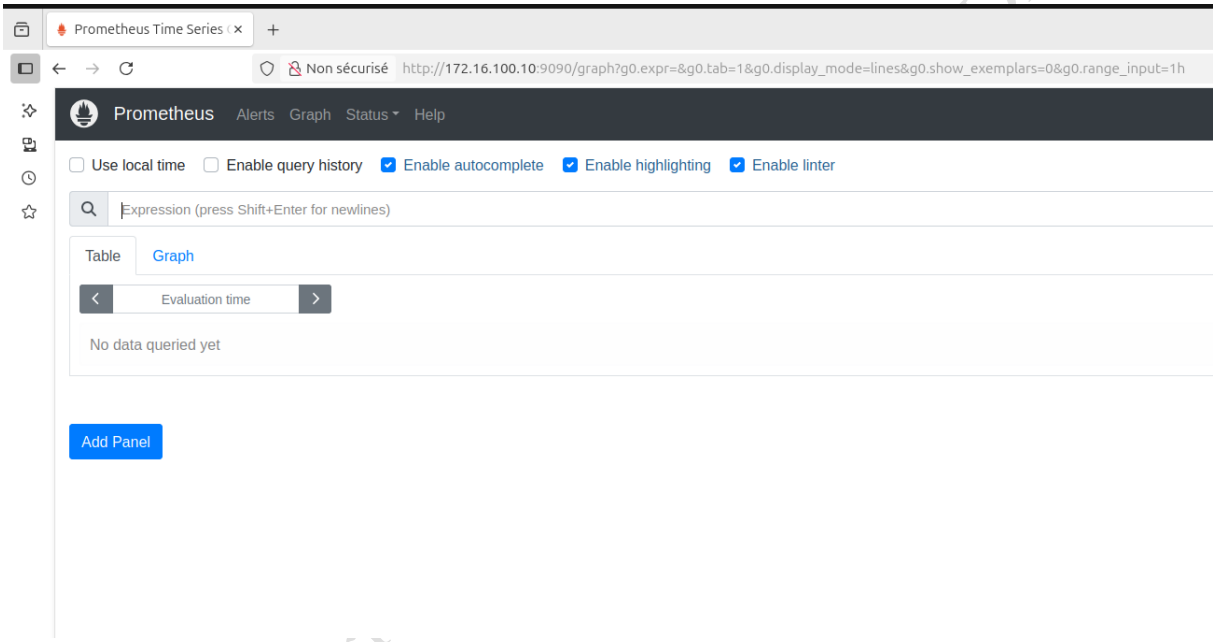
Mettre la configuration suivante

```
root@rassoul-Zabbix: /home/rassoul/prometheus-2.55.0.linux-amd64
[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target

[Service]
User=prometheus
Group=prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
  --config.file=/etc/prometheus/prometheus.yml \
  --storage.tsdb.path=/var/lib/prometheus/ \
  --web.console.templates=/etc/prometheus/consoles \
  --web.console.libraries=/etc/prometheus/console_libraries

[Install]
WantedBy=multi-user.target
```

Activer et démarrer le service



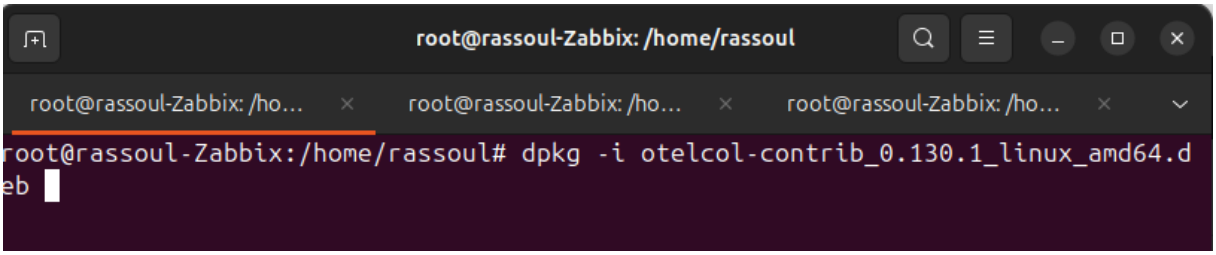
Accéder à l'interface web de Prometheus **http://mon_ip:9090**



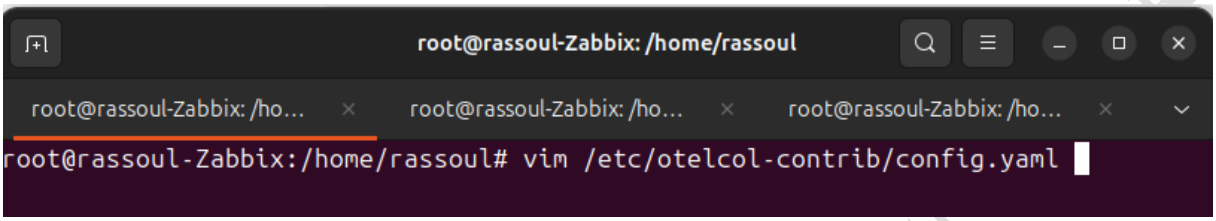
Installation de OpenTelemetry

Télécharger le fichier d'installation

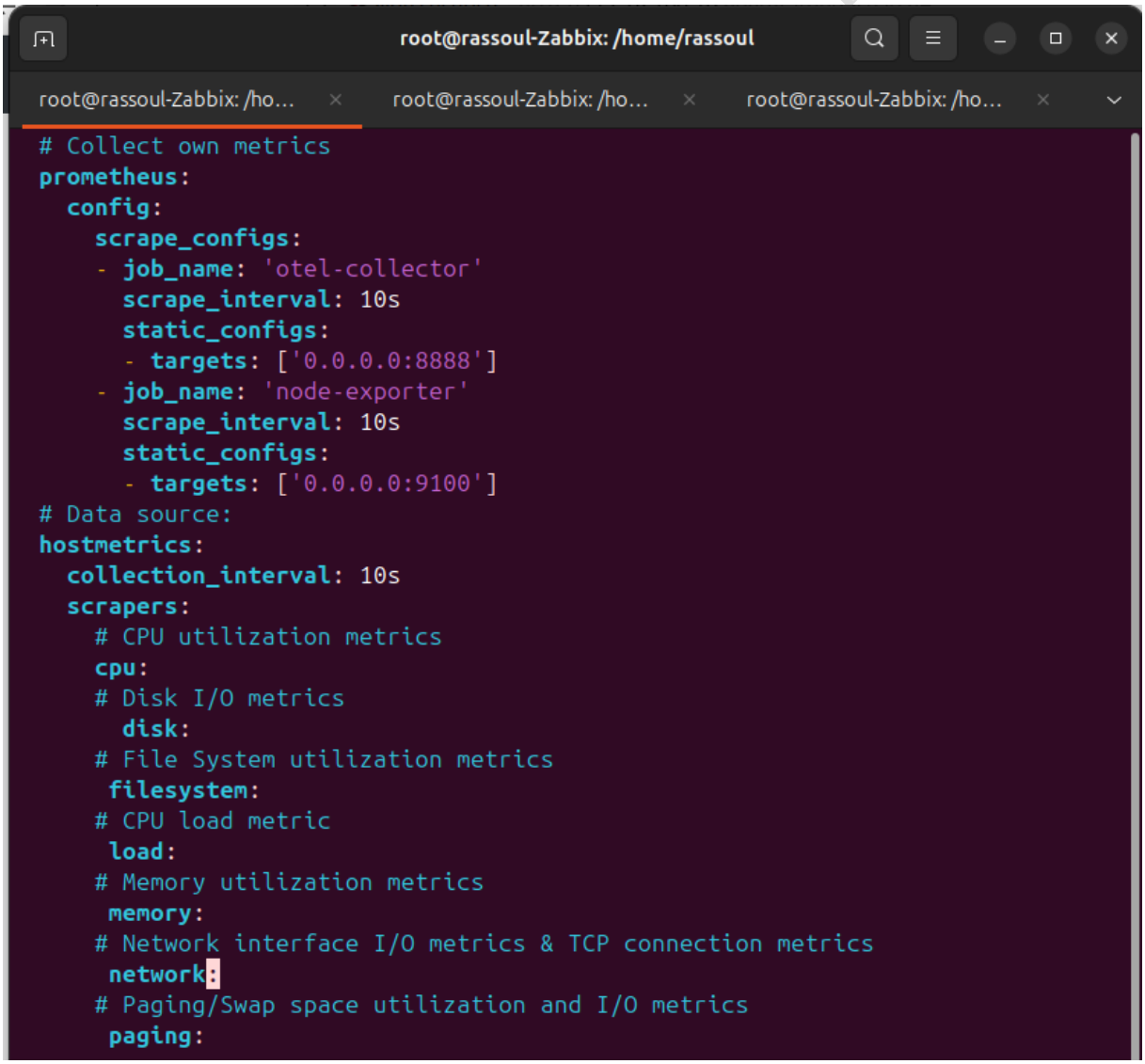
Installer comme suit

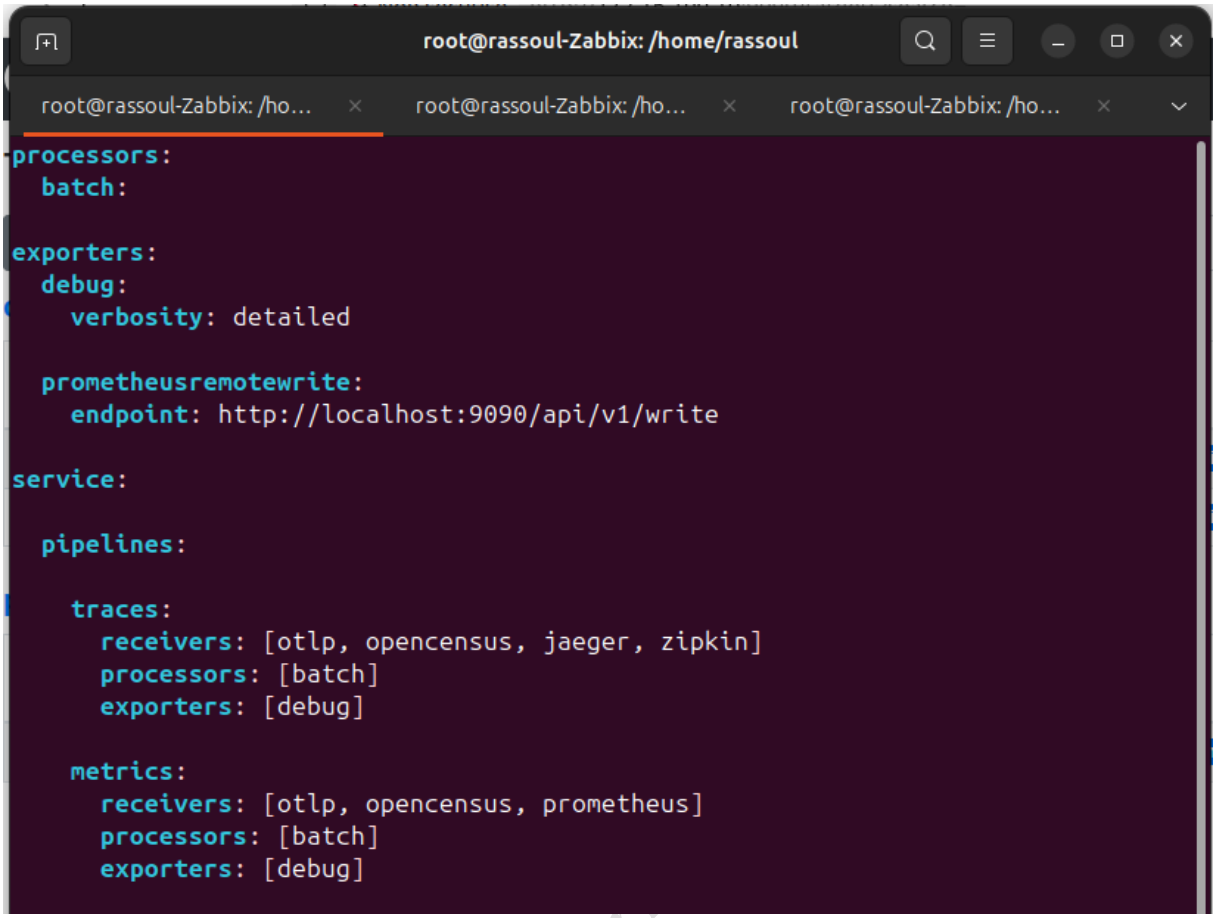


Editer le fichier de configuration

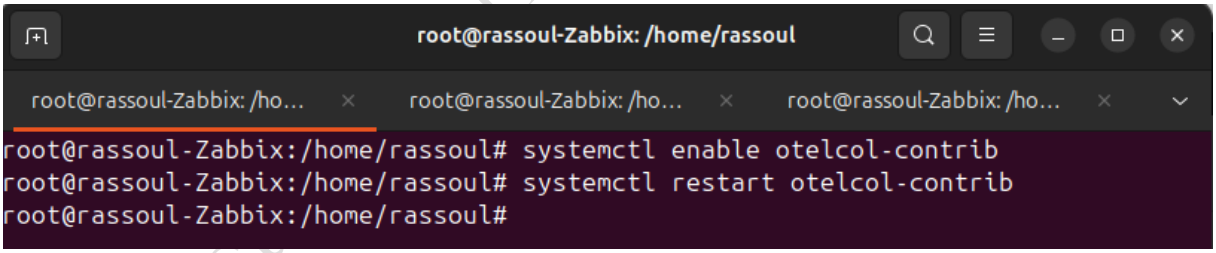


Mettre la configuration suivante





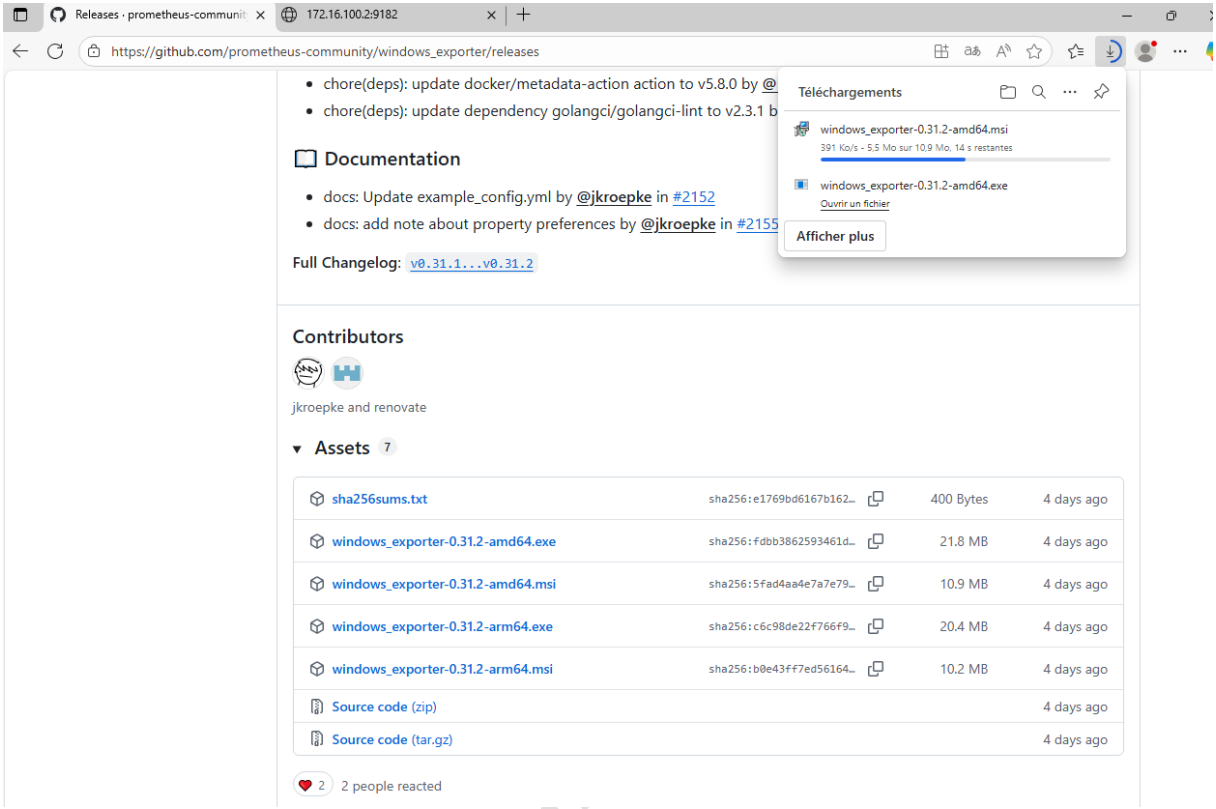
Activer et redémarrer OpenTelemetry

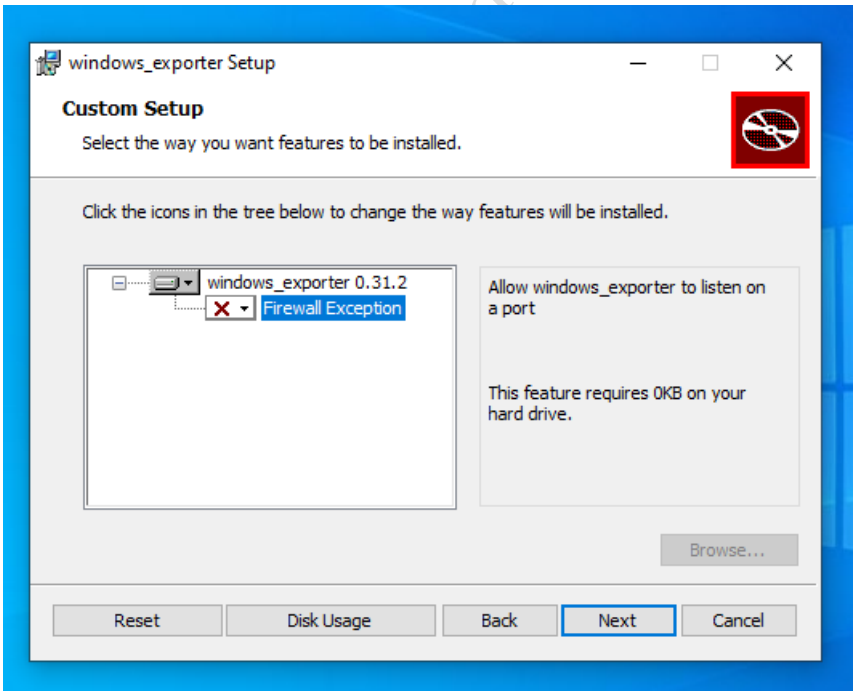


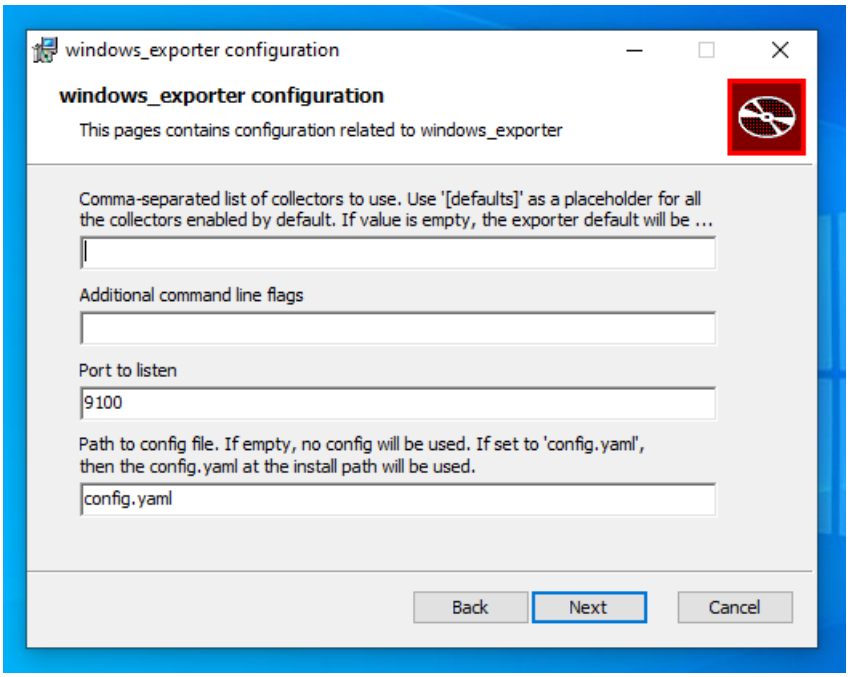
Institut Supérieur d'Informatique

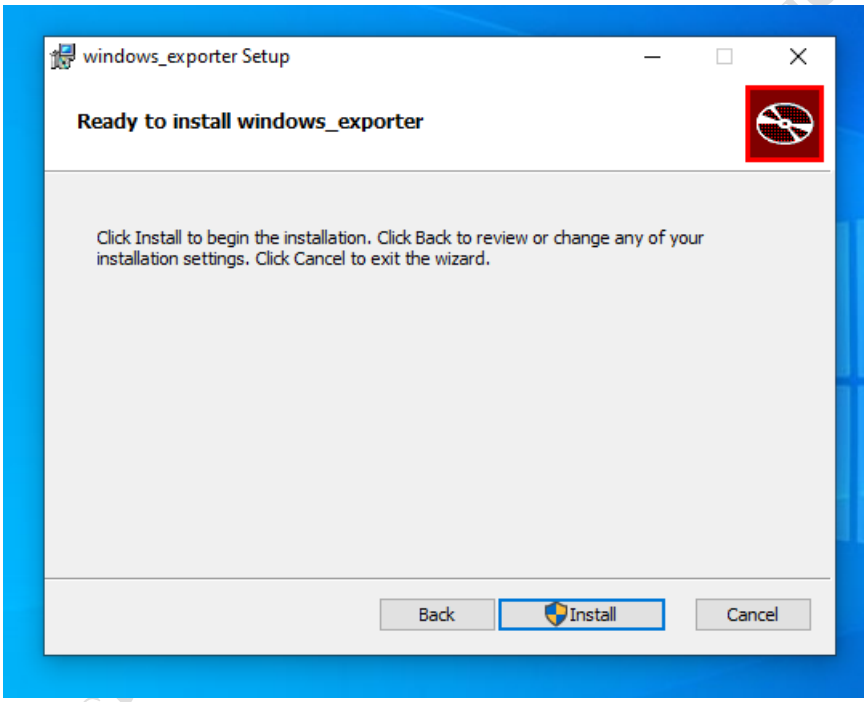
VII. Installation et configuration des agents sur Windows

Télécharger et installer **Exporter** pour windows comme suit



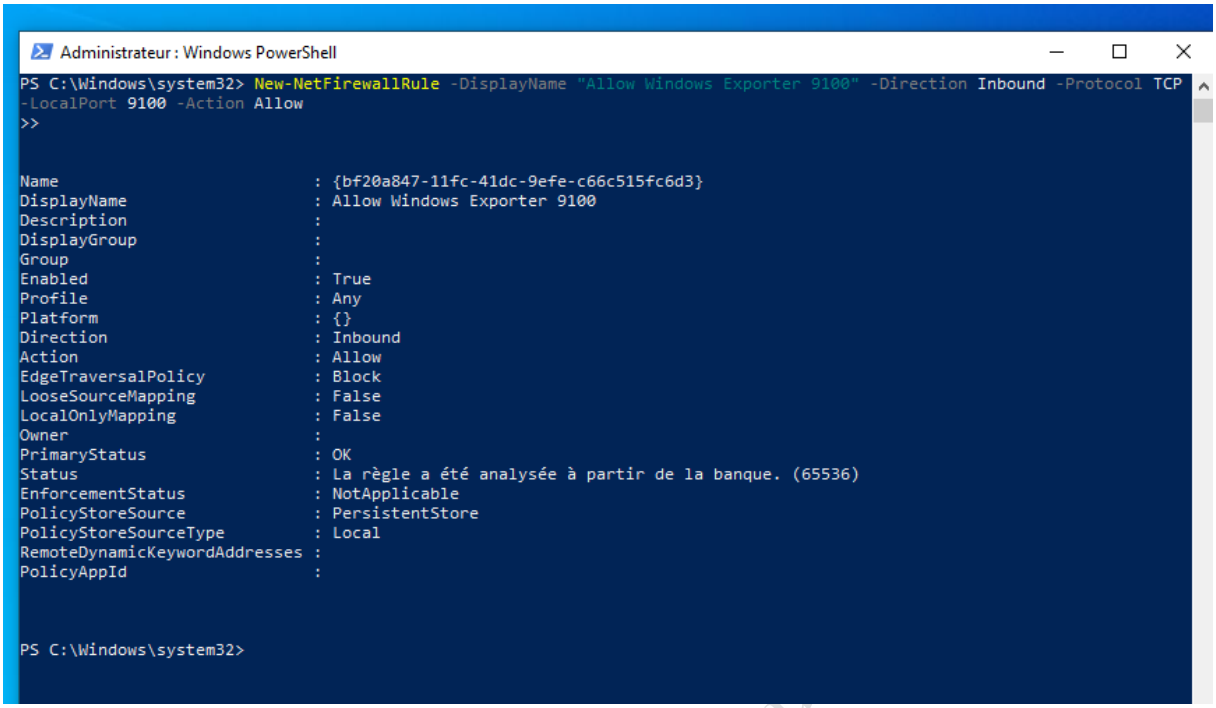






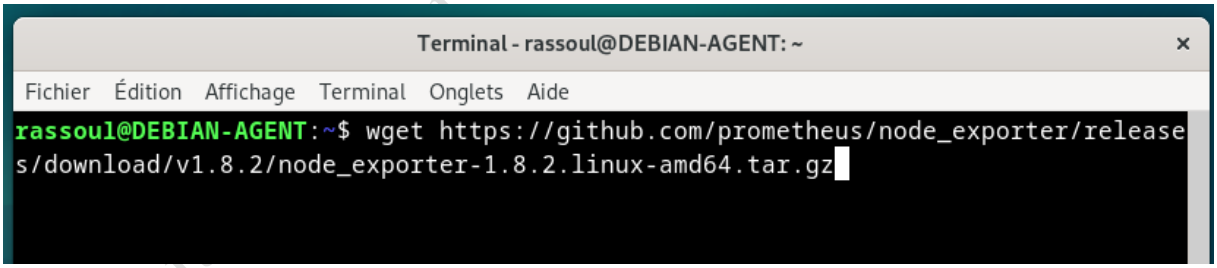
Institut Supérieur d'Informatique - ISI DAKAR

Autoriser l'écoute sur le **port 9100** en modifiant les règles du pare-feu de windows

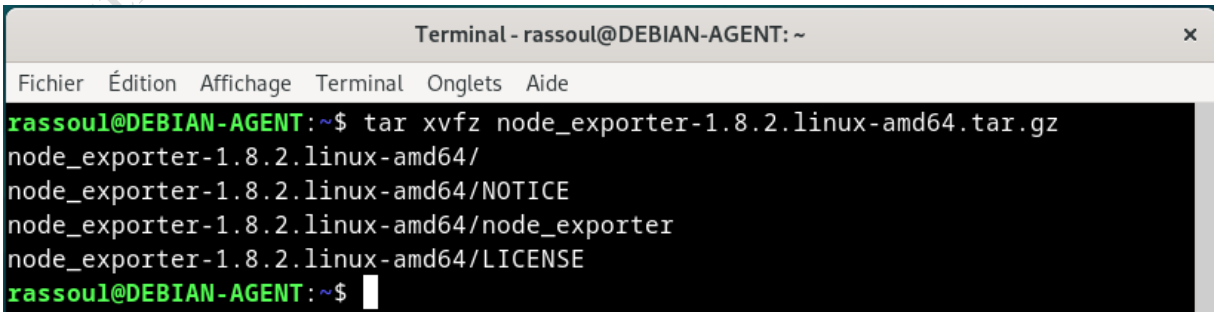


VIII. Installation et configuration des agents sur Debian

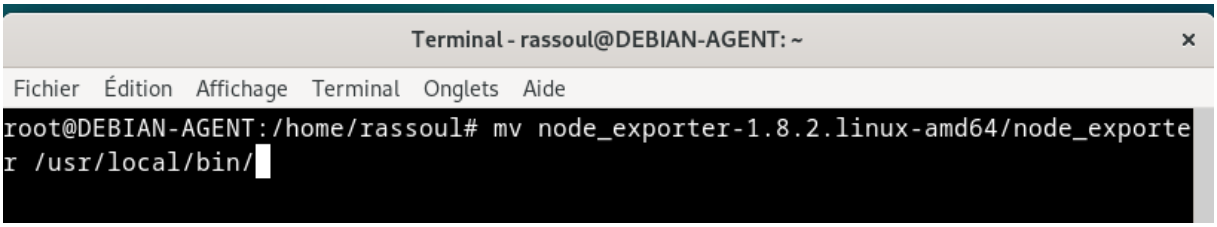
Télécharger **Exporter** pour Linux

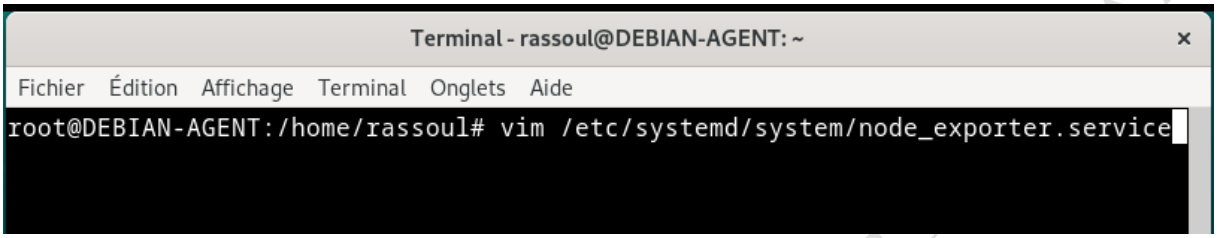


Désarchiver le fichier

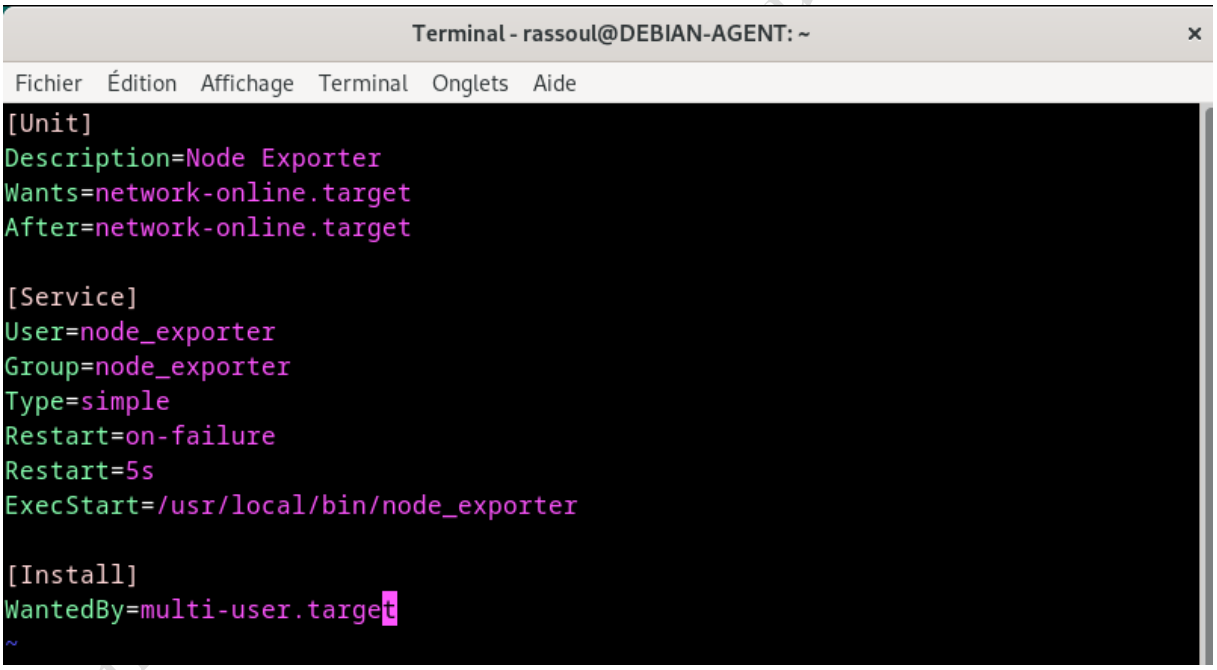


Copier le dossier créer l'utilisateur pour le service avec un shell null, et créer le fichier de configuration du service **Node exporter**

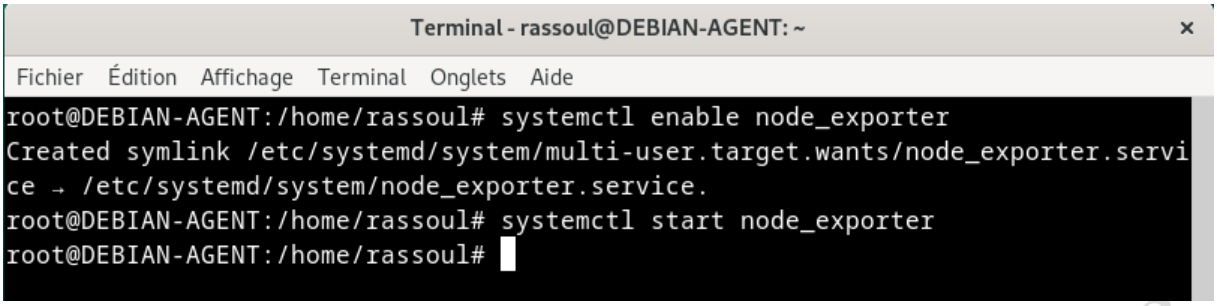




Mettre la configuration suivante



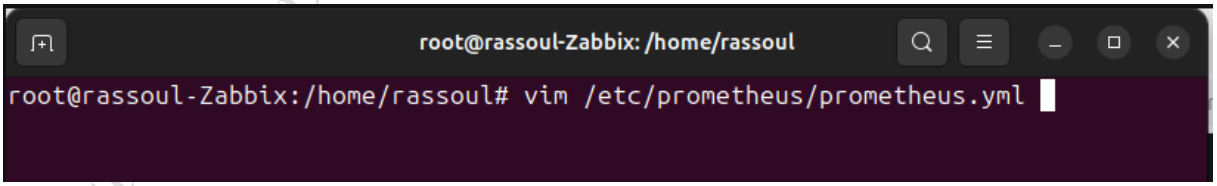
Activer et démarrer le service



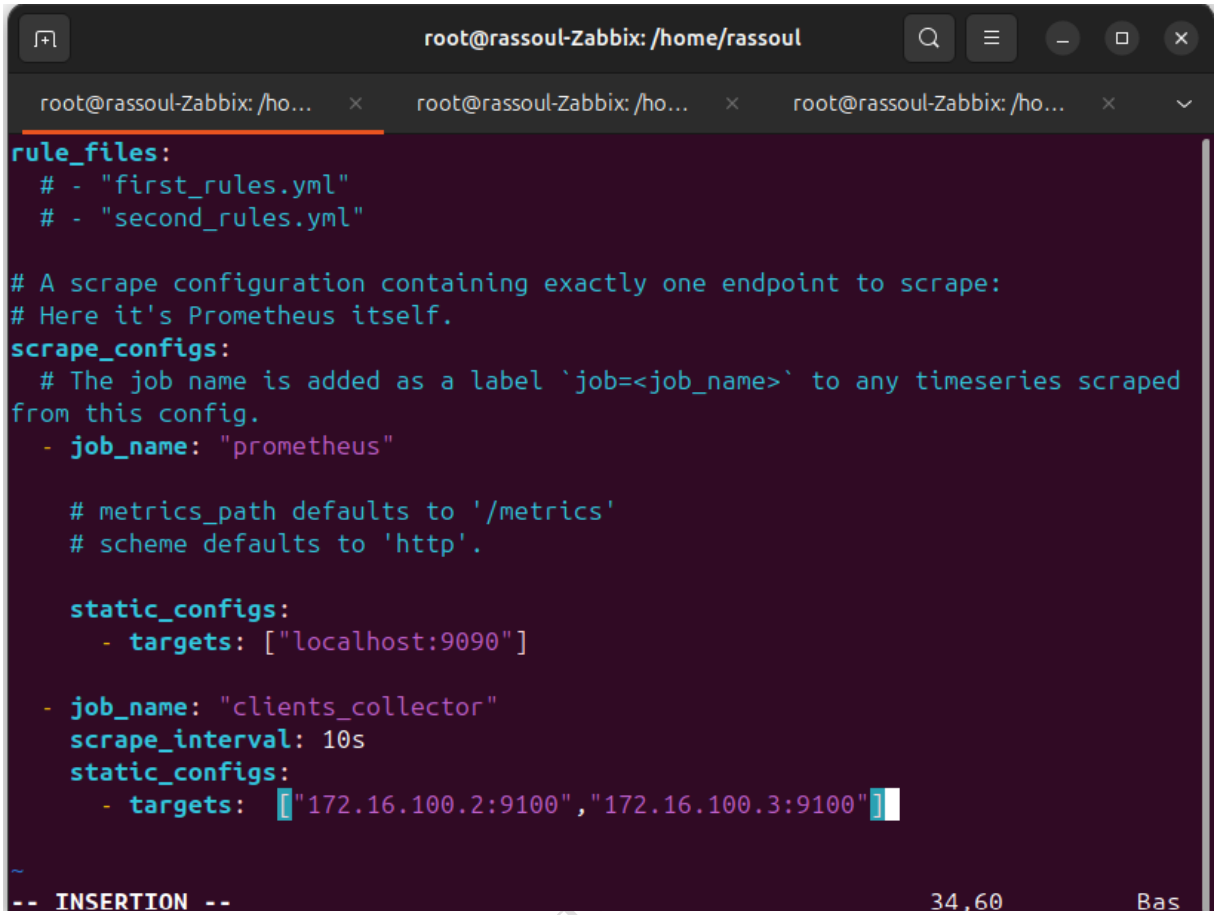
Vérifier le bon fonctionnement à travers l'interface web **http://mon_ip:9100**

Sur le serveur UBUNTU

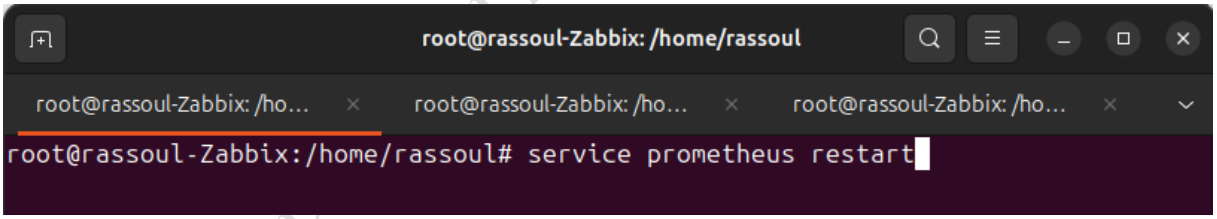
Modifier le fichier de configuration de Prometheus



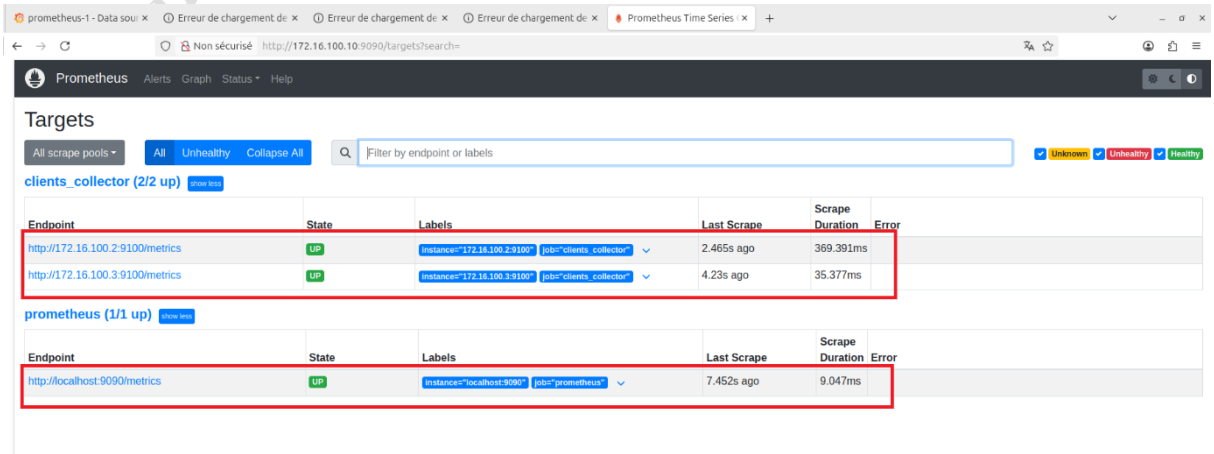
Ajouter les deux agents comme suit



Redémarrer le service Prometheus

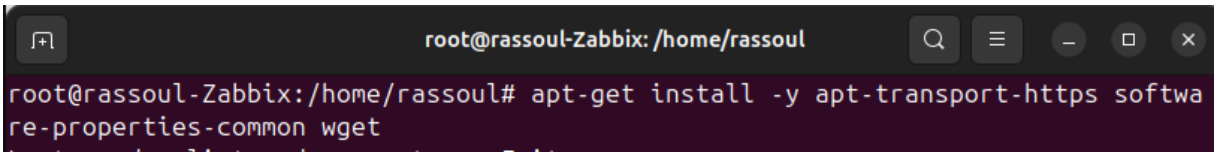


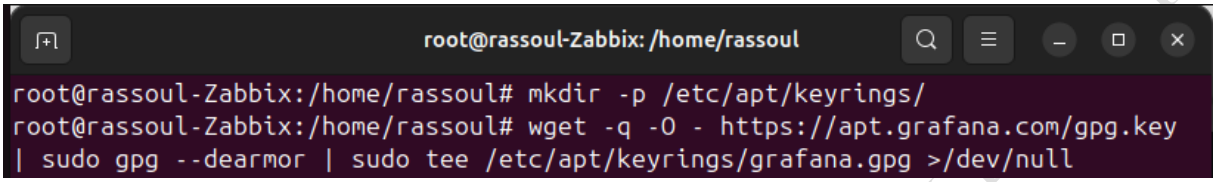
Accéder à l'interface web de Prometheus et dans **Status>Targets** vérifier la présence des deux agents

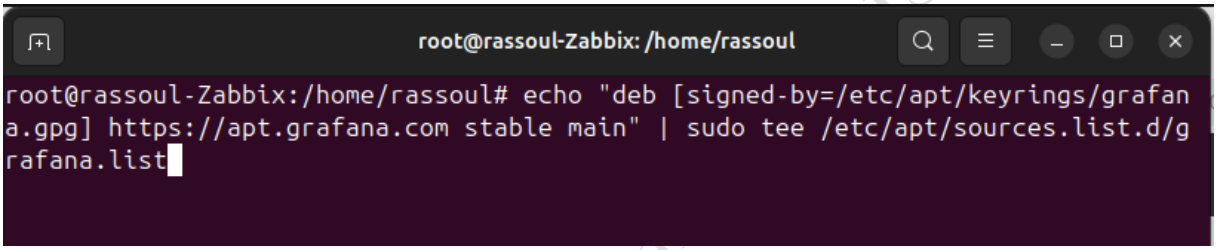


IX. Intégration avec Grafana

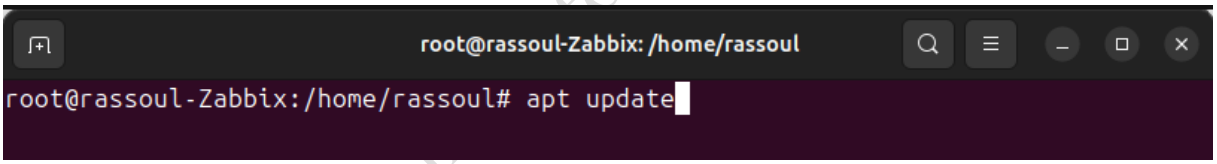
Installer les dépendances suivantes en intégrant les clés de signature



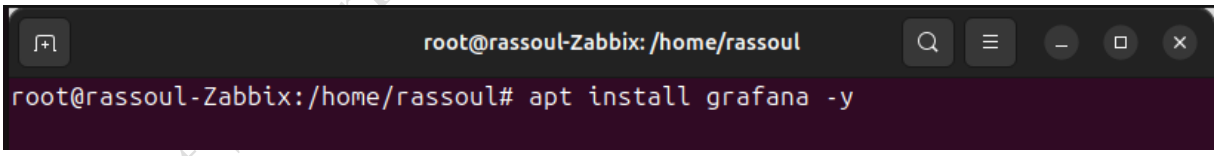




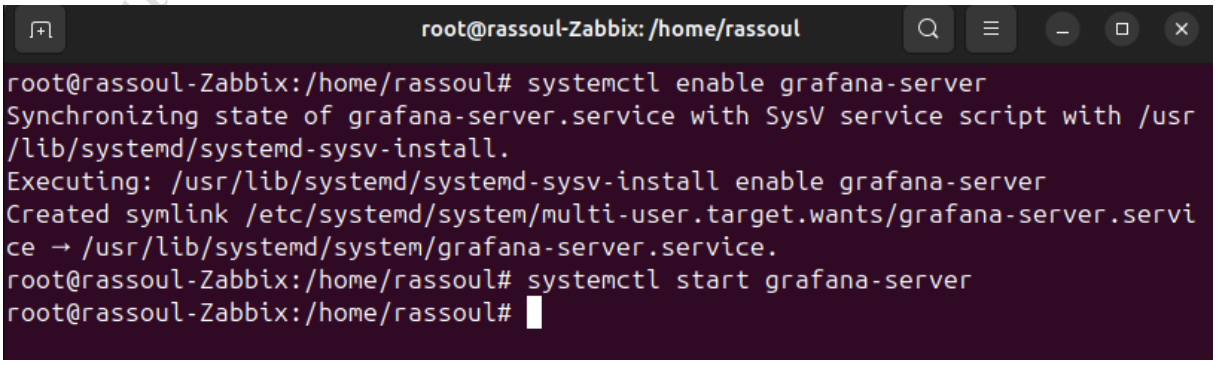
Mettre à jour la base APT



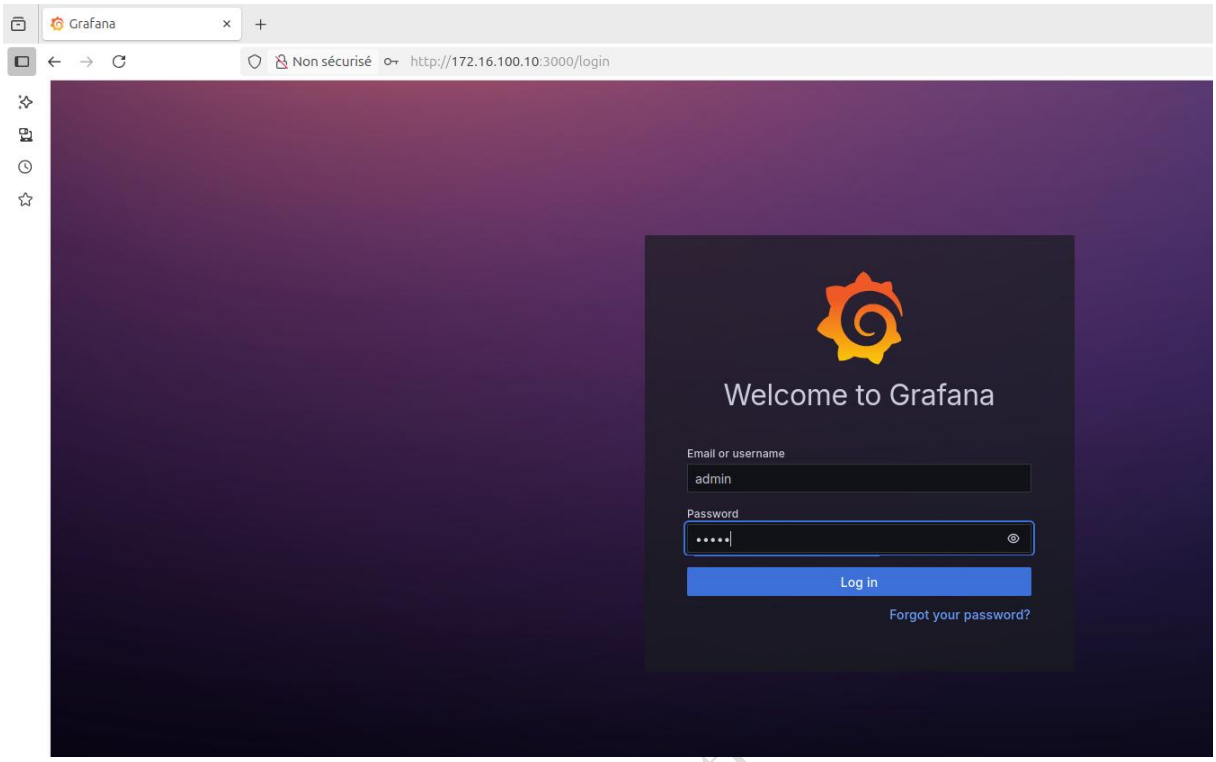
Installer **Graffana**



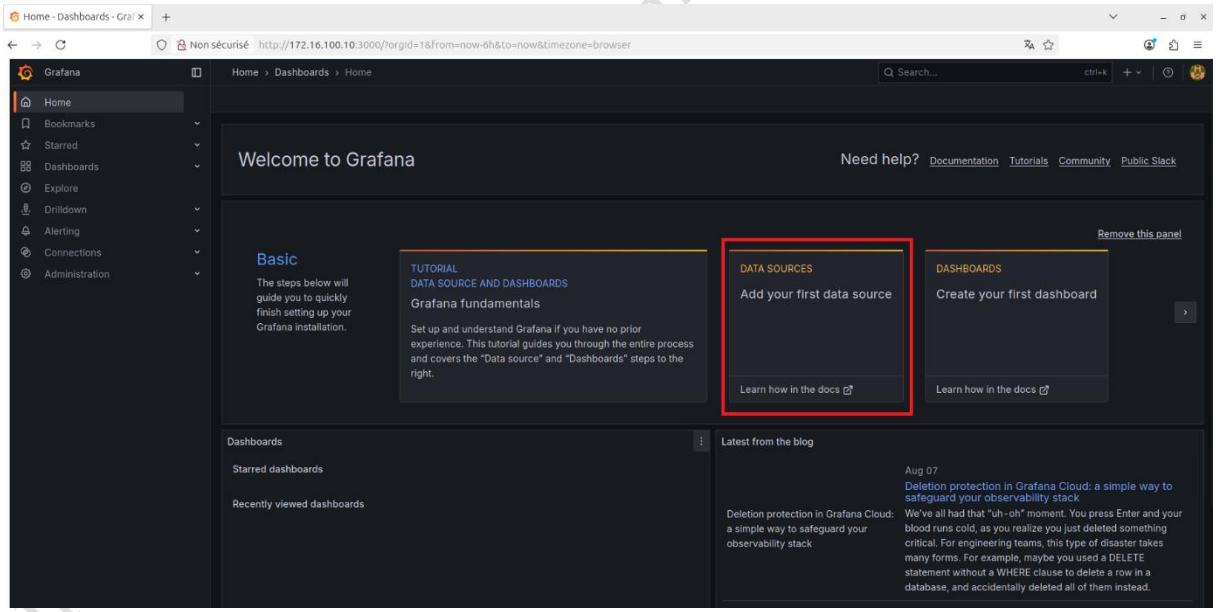
Activer et démarrer le service Grafana



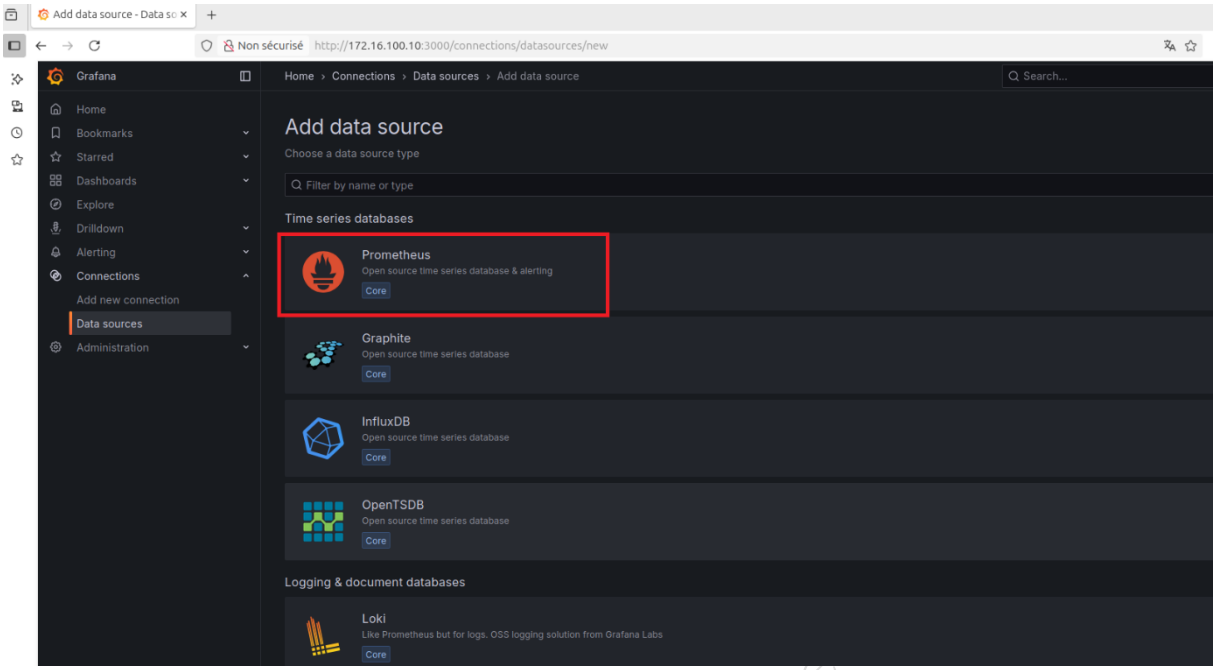
Accéder à l'inface web de Grafana **http://mon_ip:3000**



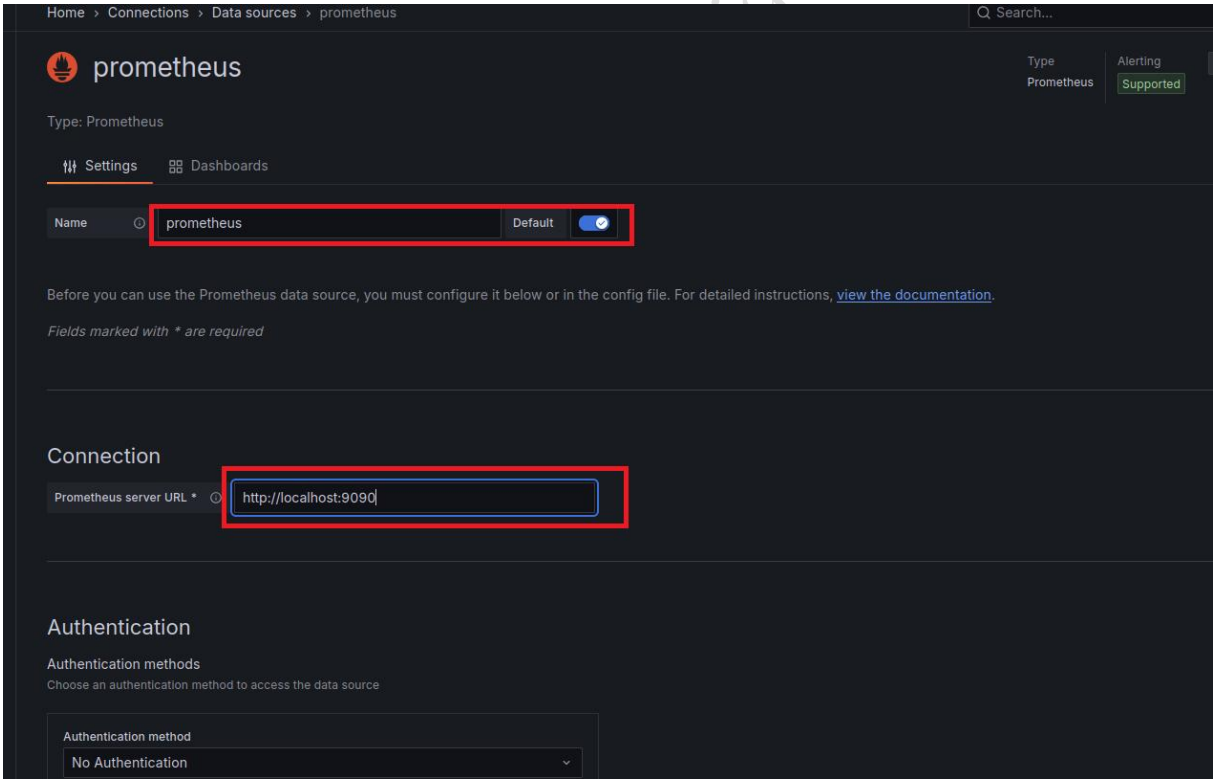
Ajouter une source de données



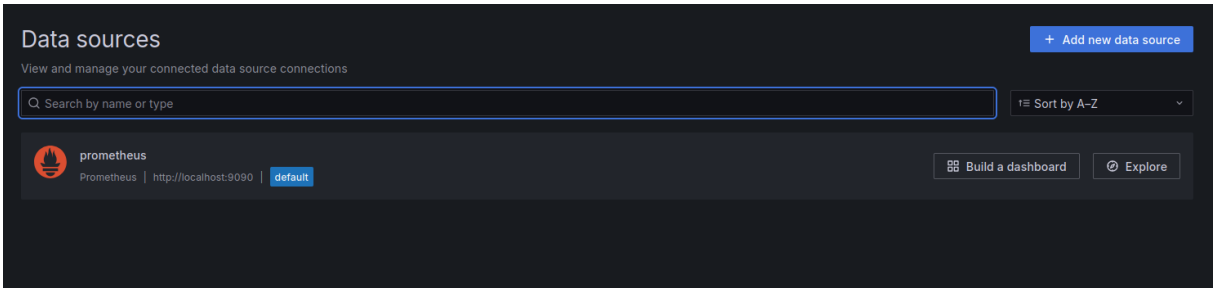
Sélectionner **Prometheus**



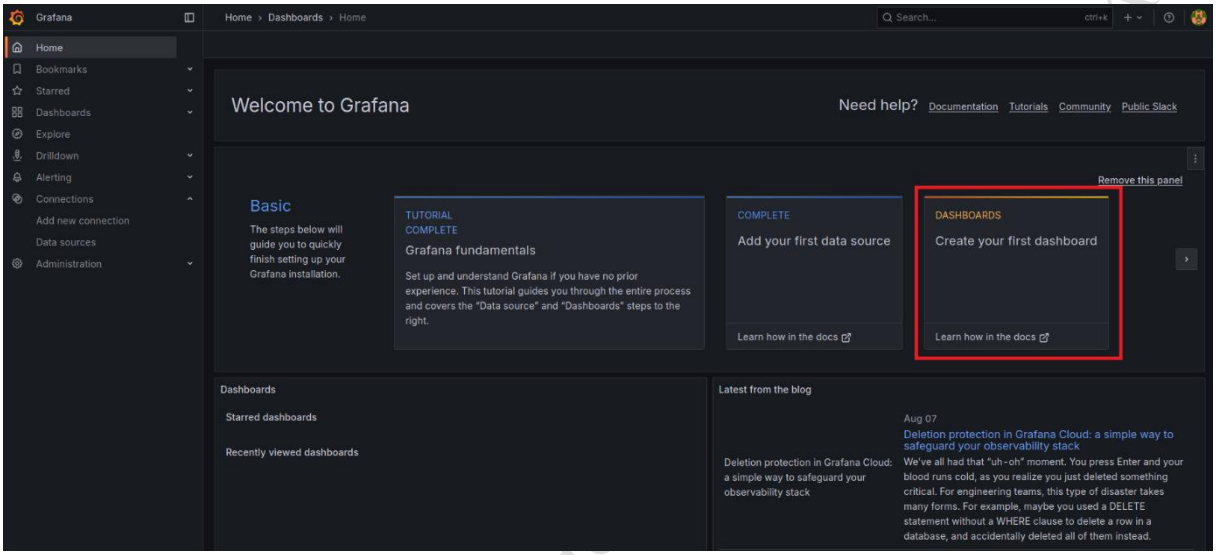
Renseigner comme suit

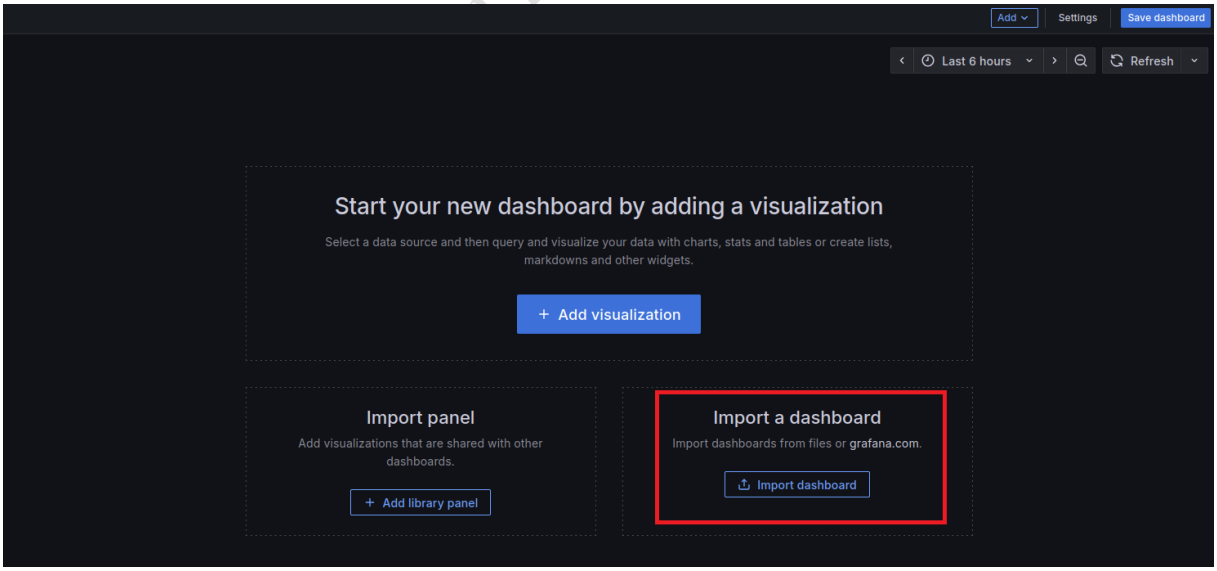


La source a bien été ajoutée

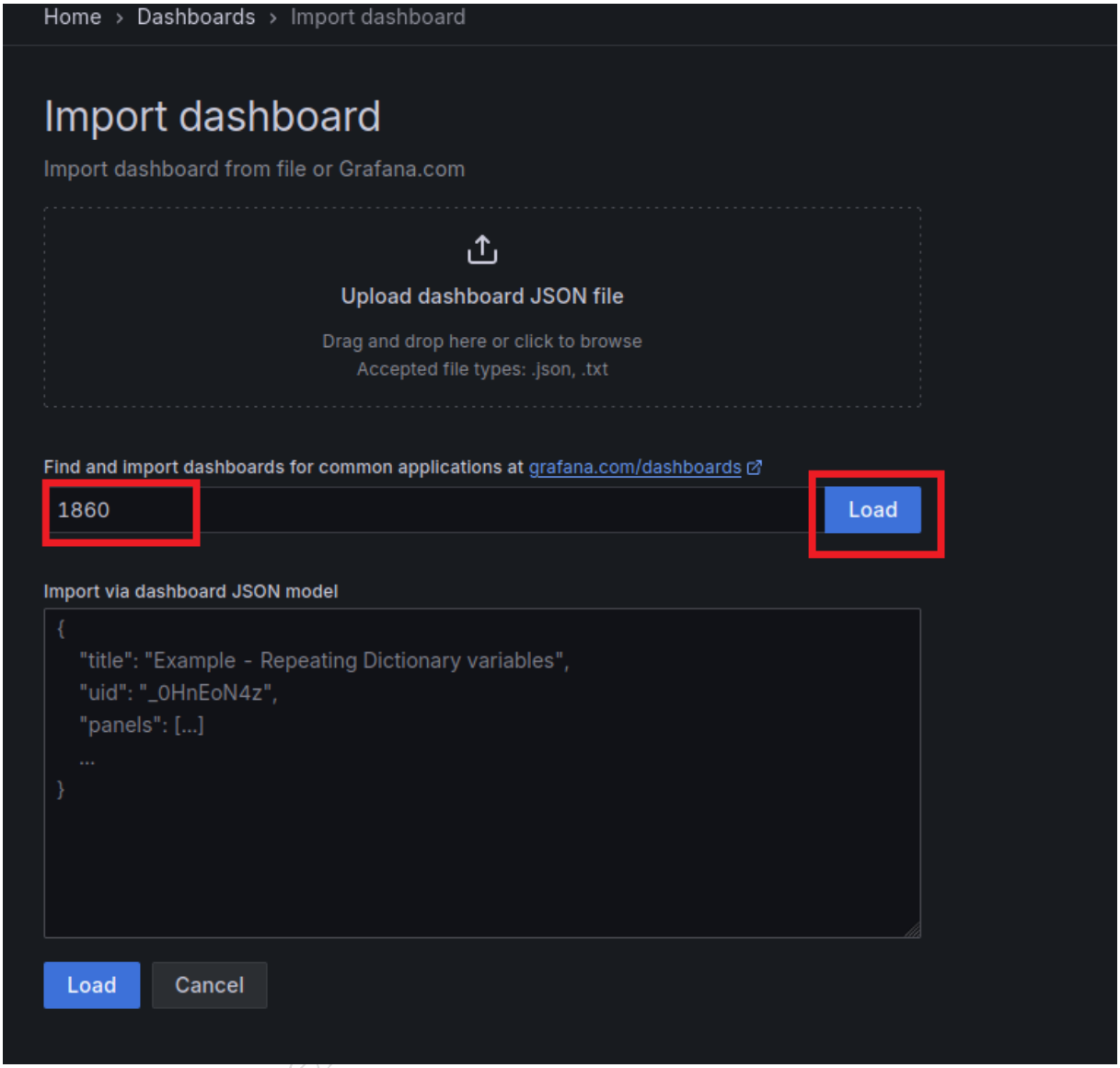


Ajouter un dashbord pour visualiser les agents



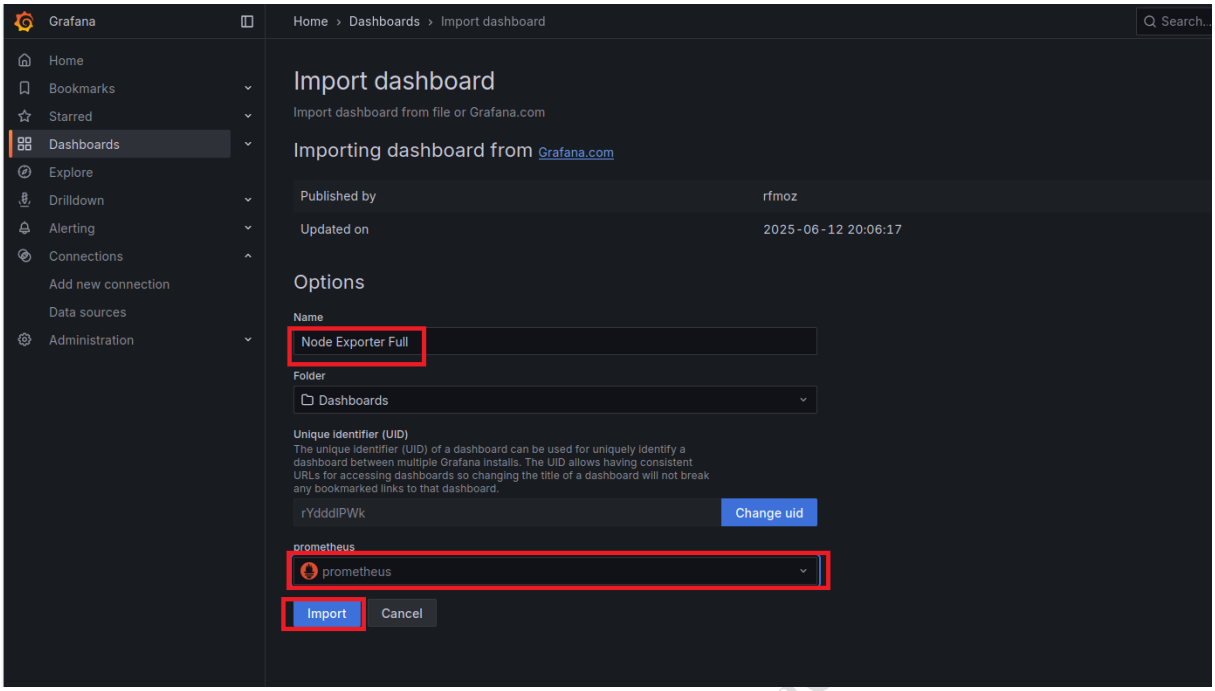


Renseigner **l'UID** et charger **1860** pour Linux (Voir sur le site officiel au besoin)

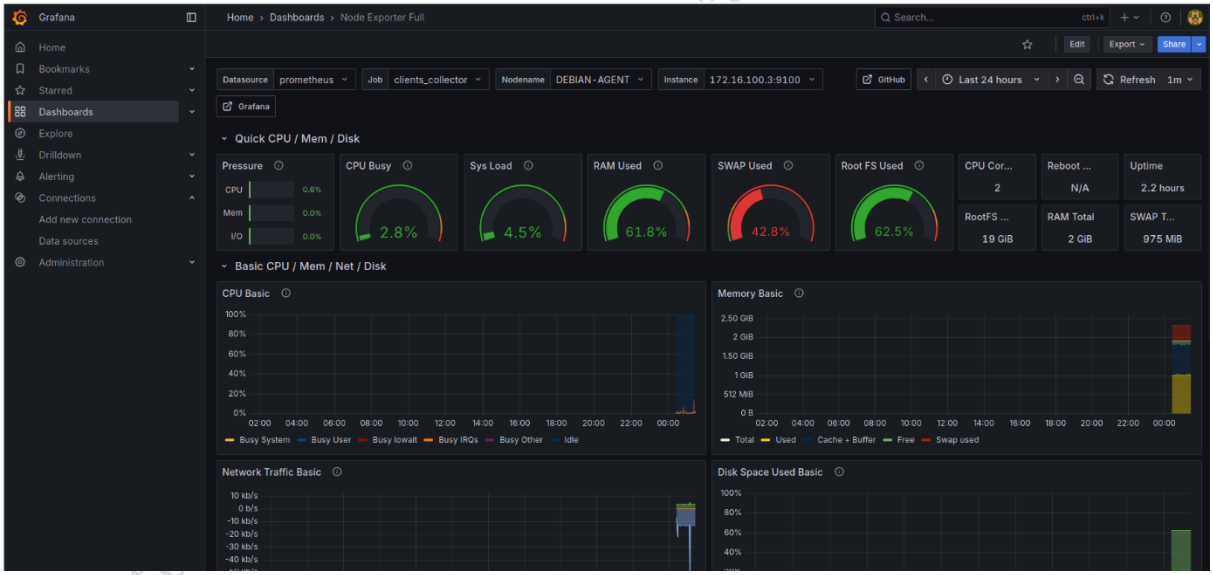


Institut Supe

Sélectionner Prometheus comme source et importer



Nous pouvons maintenant visualiser l'agent Debian



X. Analyse des données collectées

Windows : suivi de l'usage CPU et RAM, alertes sur disque à 85%, pics réseau lors de transferts massifs.

Debian : charge système stable, faible consommation mémoire, monitoring réseau constant.

Corrélations : association des données systèmes avec les logs applicatifs via OpenTelemetry pour détecter la cause des anomalies.

XI. Conclusion

Cette infrastructure de supervision et d'observabilité offre une vision globale et temps réel de la santé du système. Grâce à la flexibilité d'OpenTelemetry, à la robustesse de Prometheus et à la richesse visuelle de Grafana, l'environnement mis en place est capable de prévenir les incidents, d'optimiser les performances et de soutenir une croissance de l'infrastructure sans perte de visibilité.

Institut Supérieur d'Informatique - ISI DAKAR